

Figure 1a. Conventional Certificate Signing Process

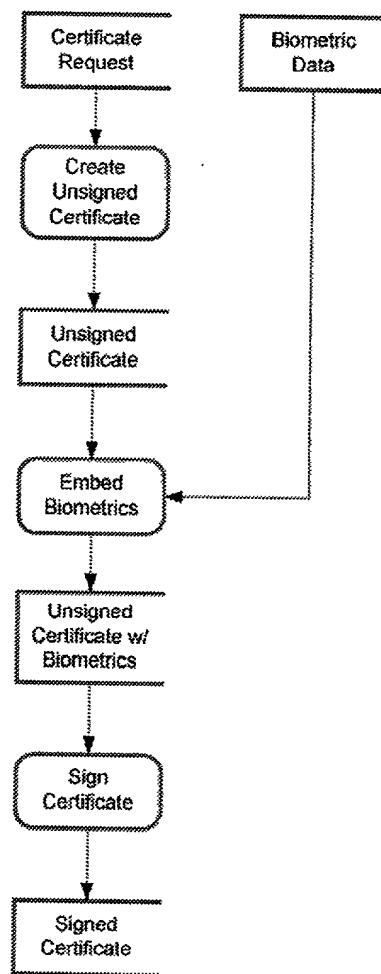


Figure 1b. Modified Certificate Signing Process

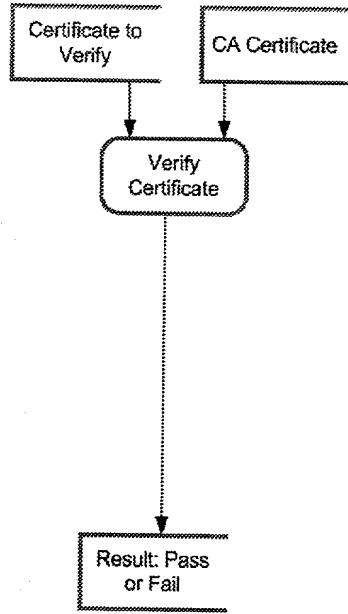


Figure 2a. Conventional
Certificate Verification
Process

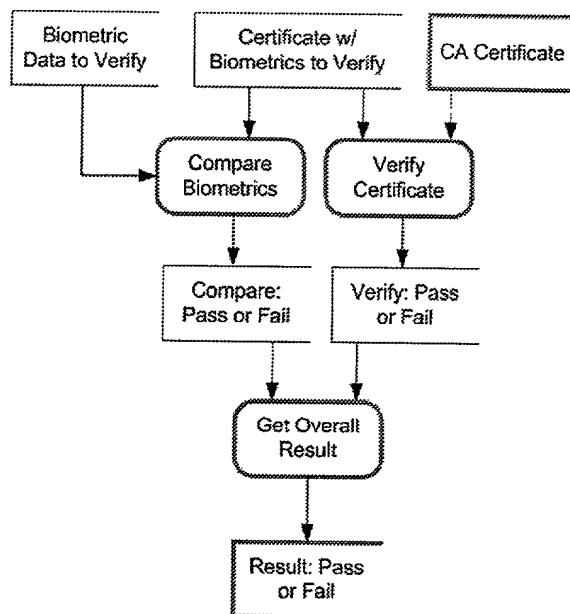


Figure 2b. Modified
Certificate Verification
Process

Version
Serial Number
Algorithm Identifier - Algorithm - Parameters
Issuer (CA)
Period of Validity - Not Before - Not After
Subject (Certificate Holder)
Subject's Public Key - Algorithm - Parameters - Public Key
Signature (by CA)

Figure 3a. Standard X.509 Certificate Structure

Version
Serial Number
Algorithm Identifier - Algorithm - Parameters
Issuer (CA)
Period of Validity - Not Before - Not After
Subject (Certificate Holder)
Subject's Public Key - Algorithm - Parameters - Public Key
Biometric Data Block
Signature (by CA)

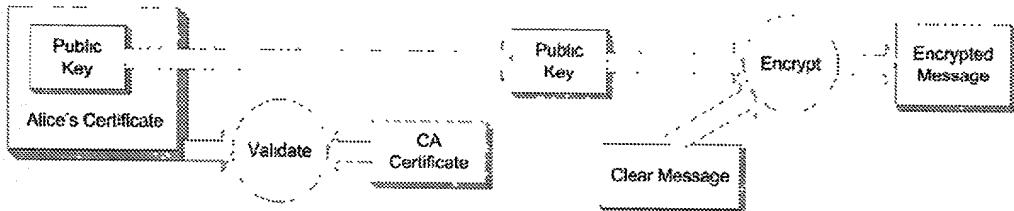
Figure 3b. Standard X.509 Certificate Structure with Embedded Biometric Data

Descriptive Header - Biometric Data Type - Location of Data - Encoding Format
Hash of Encoded Data
Encoded Biometric Data

Figure 4a. Encoding of Biometric Data Block - Biometric Data Embedded

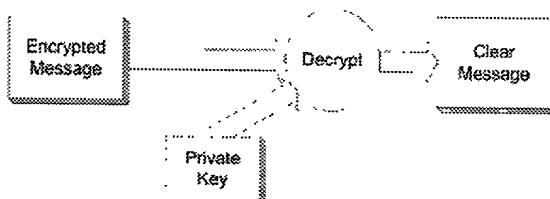
Descriptive Header - Biometric Data Type - Location of Data - Encoding Format
Hash of Encoded Data at Location

Figure 4b. Encoding of Biometric Data Block - Biometric Data Referenced



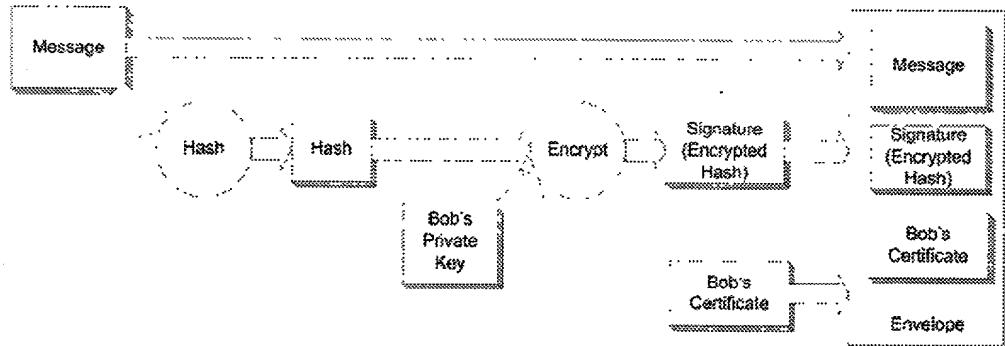
Alice gives Bob her certificate.	Bob validates Alice's certificate against the CA's certificate.	Bob uses the public key from Alice's certificate	and a clear text message,	as input to an encryption engine, to produce	an encrypted message that only Alice can read.
----------------------------------	---	--	---------------------------	--	--

Figure 5a. Using public key encryption to encrypt a message



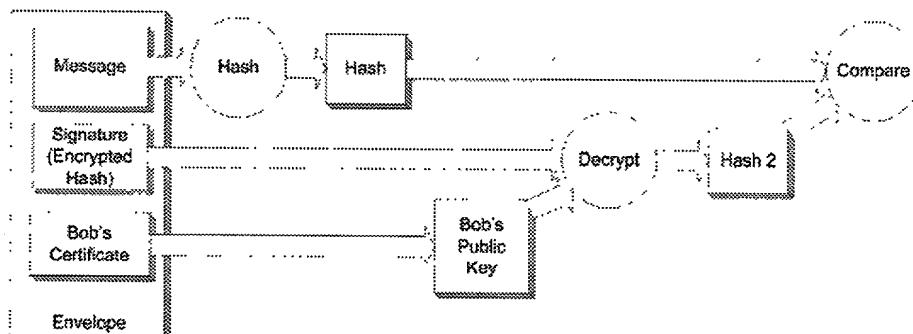
Alice uses the encrypted message from Bob,	and her private key,	as input to an decryption engine, to produce	the clear text message from Bob.
--	----------------------	--	----------------------------------

Figure 5b. Using public key encryption to decrypt a message



Bobs uses his message as input to a hash engine to produce a hash. He then uses this hash, and his private key, as input to an encryption engine, to produce a signature. Bobs puts the message, the hash, and his certificate into a digital 'envelope'.

Figure 6a. Using public key encryption to sign a message



Alice uses Bob's message as input to a hash engine to produce a hash. She then uses the public key from Bob's certificate, and Bob's signature as input to a decryption engine, to produce another hash. If the hashes match, she knows she has a valid signature.

Figure 6b. Using public key encryption to validate a digital signature

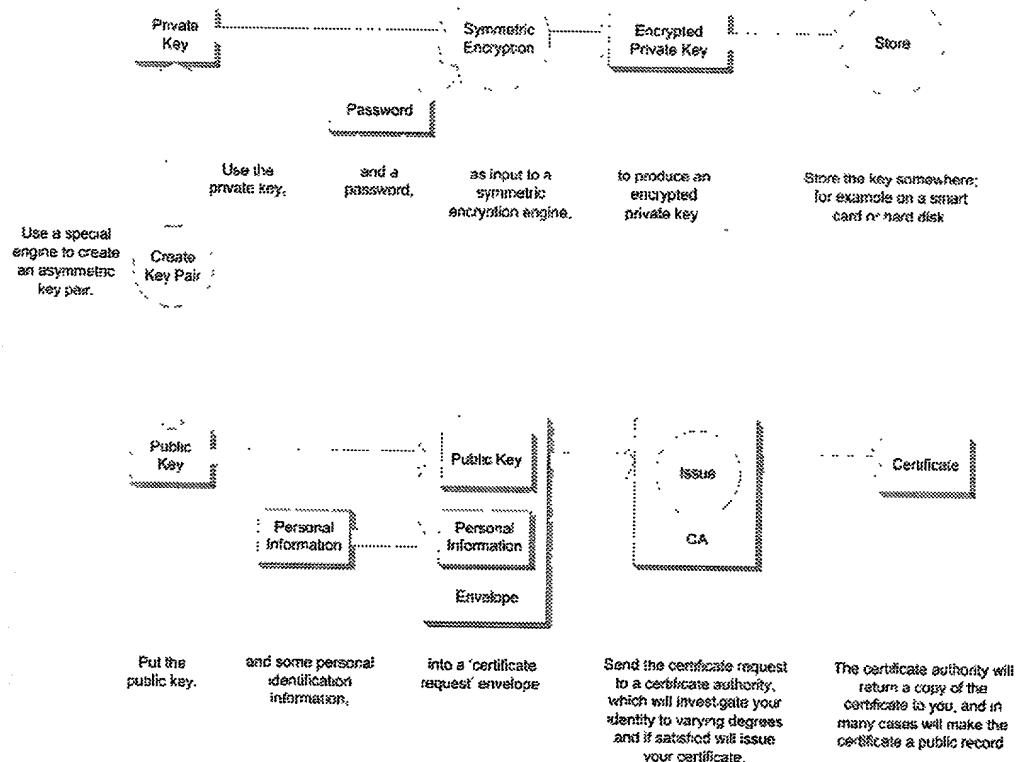


Figure 7. Using public key encryption to acquire a signed digital certificate